Claims

1. An accelerated graphics processing subsystem comprising:

a graphics command replicator;

a plurality of off-the-shelf video cards;

a mechanism to synchronize the signal by said plurality of video cards; and

a video merger hub.

2. The accelerated graphics processing subsystem of Claim 1 wherein:

said graphics command replicator is a software module that intercepts graphics commands issued by an application and generates multiple, modified graphics command streams;

the number of said multiple, modified graphics command streams is equal to the number of said plurality of video cards; and

each of said multiple, modified graphics command streams is received by a separate video card selected from said plurality of video cards.

3.   The accelerated graphics processing subsystem of Claim 2
wherein said graphics command replicator generates said multiple,
modified graphics command streams such that each of said multiple,
modified graphics command streams contains commands to draw only a

5   portion of a graphics screen.

4.   The accelerated graphics processing subsystem of Claim 3
wherein the output signals from said plurality of video cards is combined
into a single graphics output signal by said video merger hub and is
displayed on a visual output device.

10   5.   The accelerated graphics processing subsystem of Claim 4
wherein said visual output device is an output device selected from the
group consisting of: cathode ray tube displays, liquid crystal displays,
plasma screen displays, projection displays, OLED displays, head-
mounted displays and hybrids thereof.

15   6.   The accelerated graphics processing subsystem of Claim 2
wherein said intercepted graphics commands are API commands and said
multiple, modified graphics command streams are multiple, modified API
command streams.

7.   The accelerated graphics processing subsystem of Claim 6
20   wherein said multiple, modified API command streams are each received

by a separate instance of an API module and wherein each of said API module instances generates a command stream which is processed by a separate video card selected from said plurality of video cards.

8.   The accelerated graphics processing subsystem of Claim 4 wherein said modifications to said multiple, modified graphics command streams are accomplished by incorporating therein a clipping command.

9.   The accelerated graphics processing subsystem of Claim 8 wherein said clipping command is a 2D clipping command.

10.   The accelerated graphics processing subsystem of Claim 8 wherein said clipping command is a 3D clipping command.

11.   The accelerated graphics processing subsystem of Claim 4 wherein the sum of all said portions of said graphics screen combine to generate a full graphics screen.

12.   The accelerated graphics processing subsystem of Claim 11 wherein said portions of said graphics screen are non-overlapping.

13.   The accelerated graphics processing subsystem of Claim 11 wherein said portions of said graphics screen have overlapping regions.

14.   The accelerated graphics processing subsystem of Claim 4 wherein the sum of all said portions of said graphics screen combine to generate a partial graphics screen.

15.   The accelerated graphics processing subsystem of Claim 14 wherein said portions of said graphics screen are non-overlapping.

16.   The accelerated graphics processing subsystem of Claim 14 wherein said portions of said graphics screen have overlapping regions.

17.   The accelerated graphics processing subsystem of Claim 4 wherein each of said plurality of video cards is equipped with a single GPU.

18.   The accelerated graphics processing subsystem of Claim 4 wherein each of said plurality of video cards is equipped with a plurality of GPUs.

19.   The accelerated graphics processing subsystem of Claim 4 wherein said plurality of video cards is comprised of a combination of video cards equipped with a plurality of GPUs and video cards equipped with a single GPU.

20. The accelerated graphics processing subsystem of Claim 4 wherein said mechanism to synchronize the signal output by said plurality of video cards is a genlock mechanism.

21. The accelerated graphics processing subsystem of Claim 4 wherein said mechanism to synchronize the signal output by said plurality of video cards consists of designating the timing regulating device in one of said plurality of video cards as a master timing regulating device and designating timing regulating devices in the remainder of said plurality of video cards as slaves of said master timing regulating device.

22. The accelerated graphics processing subsystem of Claim 21 wherein the timing reference sources for said master and slave timing regulating devices are timing reference sources selected from the group consisting of piezoelectric crystals, programmable crystals, oscillators, programmable oscillators and combinations thereof.

23. The accelerated graphics processing subsystem of Claim 4 wherein said video merger hub is comprised of:

a video switch;

a video switch controller;

a microcontroller; and

a video output.

24.   The accelerated graphics processing subsystem of Claim 23 wherein said video switch receives said output signals from said plurality of video cards and sequentially routes selected portions of said output signals from said plurality of video cards to said video output.

25.   The accelerated graphics processing subsystem of Claim 24 wherein said video switch is controlled by said video switch controller.

26.   The accelerated graphics processing subsystem of Claim 25 wherein said video switch controller controls said video switch by triggering routing switches at appropriate intervals.

27.   The accelerated graphics processing subsystem of Claim 26 wherein said routing switch triggering intervals are determined on the basis of:

the vertical refresh rate of said output signals from said plurality of video cards;

the vertical resolution of said output signals from said plurality of video cards; and

the load balancing ratio assigned to each of said plurality of video cards.

28. The accelerated graphics processing subsystem of Claim 27 wherein said load balancing ratio is transmitted by said microcontroller to said video switch controller.

29. The accelerated graphics processing subsystem of Claim 27 wherein said load balancing ratio is assigned by a user through software.

30. The accelerated graphics processing subsystem of Claim 27 wherein said load balancing ratio is equal for each of said video cards.

31. The accelerated graphics processing subsystem of Claim 27 wherein said load balancing ratio is based on each of said video cards' graphics throughput.

32. The accelerated graphics processing subsystem of Claim 27 wherein said load balancing ratio is dynamically adjusted to maximize the throughput of said subsystem by utilizing a test feedback loop program which measures the load on each of said video cards and makes appropriate adjustments.

33. A method for accelerating the processing of graphics instructions on a computer through use of a plurality of video cards, comprising the steps of:

intercepting graphics commands issued by an application;

generating multiple, modified graphics command streams;

5   assigning each of said multiple, modified graphics command streams to a different video cards selected from said plurality of video cards;

processing at each of said video cards the assigned modified graphics command stream and

10   generating an output signal;

synchronizing the output signals by said plurality of video cards;

combining the output signal from said plurality of video cards into a single graphics output signal;

15   and

displaying said single graphics output signal on a visual output device.

34. The method of Claim 33 wherein the output signal of each of said multiple, modified graphics command streams draws only a portion of a graphics screen.

35. The method of claim 34 wherein said intercepting and generating steps are performed by a graphics command replicator.

36. The method of claim 34 wherein said combining step is performed by a video merger hub.

37. The method of Claim 34 wherein said visual output device is an output device selected from the group consisting of: cathode ray tube displays, liquid crystal displays, plasma screen displays, projection displays, OLED displays, head-mounted displays and hybrids thereof.

38. The method of Claim 33 wherein said intercepted graphics commands are API commands and said multiple, modified graphics command streams are multiple, modified API command streams.

39. The method of Claim 38 wherein said multiple, modified API command streams are each received by a separate instance of an API module and wherein each of said API module instances generates a command stream which is processed by a separate video card selected from said plurality of video cards.

40. The method of Claim 34 wherein said generation of multiple, modified graphics command streams is accomplished by incorporating a clipping command in each stream.

41. The method of Claim 40 wherein said clipping command is a 2D clipping command.

42. The method of Claim 40 wherein said clipping command is a 3D clipping command.

43. The method of Claim 34 wherein the sum of all said portions of said graphics screen combine to generate a full graphics screen.

44. The method of Claim 43 wherein said portions of said graphics screen are non-overlapping.

45. The method of Claim 43 wherein said portions of said graphics screen have overlapping regions.

46. The method of Claim 34 wherein the sum of all said portions of said graphics screen combine to generate a partial graphics screen.

47. The method of Claim 46 wherein said portions of said graphics screen are non-overlapping.

48. The method of Claim 46 wherein said portions of said graphics screen have overlapping regions.

49. The method of Claim 34 wherein each of said plurality of video cards is equipped with a single GPU.

50. The method of Claim 34 wherein each of said plurality of video cards is equipped with a plurality of GPUs.

5   51. The method of Claim 34 wherein said plurality of video cards is comprised of a combination of video cards equipped with a plurality of GPUs and video cards equipped with a single GPU.

52. The method of Claim 34 wherein said synchronizing step is accomplished through use of a genlock mechanism.

10   53. The method of Claim 34 wherein said synchronizing step is accomplished by designating the timing regulating device in one of said plurality of video cards as a master timing regulating device and designating timing regulating devices in the remainder of said plurality of video cards as slaves of said master timing regulating device.

15   54. The method of Claim 53 wherein the timing reference sources for said master and slave timing regulating devices are timing reference sources selected from the group consisting of piezoelectric crystals, programmable crystals, and combinations thereof.

55. The method of Claim 36 wherein said video merger hub is comprised of:

a video switch;

a video switch controller;

5   a microcontroller; and

a video output.

56. The method of Claim 55 wherein said combining step further comprises:

receiving said output signals from said plurality of

10   video cards; and

sequentially routing selected portions of said output signals from said plurality of video cards to said video output.

57. The method of Claim 56 wherein said video switch is

15 controlled by said video switch controller.

58. The method of Claim 57 wherein said video switch controller controls said video switch by triggering routing switches at appropriate intervals.

59.   The method of Claim 58 wherein said routing switch triggering intervals are determined on the basis of:

the vertical refresh rate of said output signals from said plurality of video cards;

5   the vertical resolution of said output signals from said plurality of video cards; and

the load balancing ratio assigned to each of said plurality of video cards.

60.   The method of Claim 59 wherein said load balancing ratio is transmitted by said microcontroller to said video switch controller.

61.   The method of Claim 59 wherein said load balancing ratio is assigned by a user through software.

62.   The method of Claim 59 wherein said load balancing ratio is equal for each of said video cards.

63.   The method of Claim 59 wherein said load balancing ratio is based on each of said video cards' graphics throughput.

64.   The method of Claim 59 wherein said load balancing ratio is dynamically adjusted to maximize the throughput of said subsystem by

utilizing a test feedback loop program which measures the load on each of said video cards and makes appropriate adjustments.

65. An accelerated graphics processing subsystem comprising:

a graphics command replicator consisting of a software module that intercepts graphics commands issued by an application and generates multiple, modified graphics command streams;

a plurality of video cards, each equipped with one or more GPUs, wherein the number of said multiple, modified graphics command streams is equal to the number of said plurality of video cards;

a mechanism to synchronize the signal output by said plurality of video cards; and

a video merger hub comprised of a video switch, a video switch controller, a microcontroller, and a video output;

wherein

said graphics command replicator generates said multiple, modified graphics command streams

such that each of said multiple, modified graphics command streams contains commands to draw only a portion of a graphics screen;

each of said multiple, modified graphics command streams is received by a separate video graphics card selected from said plurality of video cards;

output signals from said plurality of video cards are received by said video switch and selected portions thereof are sequentially routed to said video output and displayed on a visual output device; and

said video switch is controlled by said video switch controller through the triggering of routing switches at appropriate intervals determined by the vertical refresh rate and vertical resolution of said output signals from said plurality of video cards and by the load balancing ratio assigned to each of said plurality of video cards.

66.   A method for accelerating the processing of graphics instructions on a computer through use of a plurality of video cards, comprising the steps of:

5   intercepting graphics commands issued by an application and generating multiple, modified graphics command streams wherein the number of said multiple, modified graphics command streams is equal to the number of said plurality of video cards;

10   synchronizing the signal output by said plurality of video cards;

               combining the output signal from said plurality of video cards into a single graphics output signal through use of a video merger hub comprised of a

15   video switch, a video switch controller, a microcontroller, and a video output; and

               displaying said single graphics output signal on a visual output device;

       wherein

each of said multiple, modified graphics command streams contains commands to draw only a portion of a graphics screen;

each of said multiple, modified graphics command streams is received by a separate video graphics card selected from said plurality of video cards;

output signals from said plurality of video cards are received by said video switch and selected portions thereof are sequentially routed to said video output and displayed on a visual output device; and

said video switch is controlled by said video switch controller through the triggering of routing switches at appropriate intervals determined by the vertical refresh rate and vertical resolution of said output signals from said plurality of video cards and by the load balancing ratio assigned to each of said plurality of video cards.

67.   A video merger hub for combining the output signals from a plurality of video cards into a single graphics output signal displaying said single graphics output signal on a visual output device comprised of:

a video switch;

5   a video switch controller;

a microcontroller; and

a video output.

68.   The video merger hub of Claim 67 wherein said video switch receives said output signals from said plurality of video cards and

10   sequentially routes selected portions of said output signals from said plurality of video cards to said video output.

69.   The video merger hub of Claim 68 wherein said video switch is controlled by said video switch controller.

70.   The video merger hub of Claim 69 wherein said video

15   switch controller controls said video switch by triggering routing switches at appropriate intervals.

71.   The video merger hub of Claim 70 wherein said routing switch triggering intervals are determined on the basis of:

the vertical refresh rate of said output signals from said plurality of video cards;

the vertical resolution of said output signals from said plurality of video cards; and

5   the load balancing ratio assigned to each of said plurality of video cards.

72.   The video merger hub of Claim 71 wherein said load balancing ratio is transmitted by said microcontroller to said video switch controller.

10   73.   The video merger hub of Claim 71 wherein said load balancing ratio is assigned by a user through software.

74.   The video merger hub of Claim 71 wherein said load balancing ratio is equal for each of said video cards.

75.   The video merger hub of Claim 71 wherein said load 15 balancing ratio is calculated by determining an optimized load balancing ration for each of said video cards based on its graphics throughput.

76.   The video merger hub of Claim 71 wherein said load balancing ratio is dynamically adjusted to maximize the throughput of

said subsystem by utilizing a test feedback loop program which measures the load on each of said video cards and makes appropriate adjustments.